OMEGAS: Object Mesh Extraction from Large Scenes Guided by Gaussian Segmentation

Lizhi Wang, Feng Zhou, Bo Yu, Pu Cao, and Jianqin Yin

***Abstract*—Recent advancements in 3D reconstruction technologies have paved the way for high-quality and real-time rendering of complex 3D scenes. Despite these achievements, a notable challenge persists: it is difficult to precisely reconstruct specific objects from large scenes. Current scene reconstruction techniques frequently result in the loss of object detail textures and are unable to reconstruct object portions that are occluded or unseen in views. To address this challenge, we delve into the meticulous 3D reconstruction of specific objects within large scenes and propose a framework termed OMEGAS: Object Mesh Extraction from Large Scenes Guided by GAussian Segmentation. Specifically, we propose a novel 3D target segmentation technique based on 2D Gaussian Splatting, which segments 3D consistent target masks in multi-view scene images and generates a preliminary target model. Moreover, to reconstruct the unseen portions of the target, we propose a novel target replenishment technique driven by large-scale generative diffusion priors. We demonstrate that our method can accurately reconstruct specific targets from large scenes, both quantitatively and qualitatively. Our experiments show that OMEGAS significantly outperforms existing reconstruction methods across various scenarios.**

***Index Terms*—3D reconstruction, Object segmentation, Diffusion models, Mesh generation.**

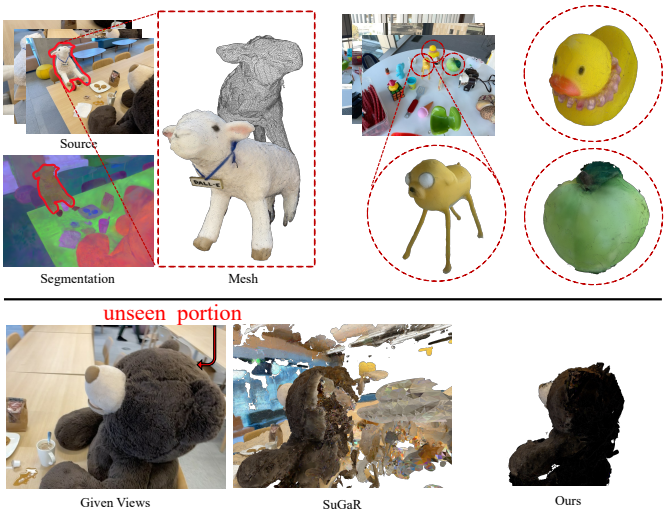


Fig. 1: OMEGAS segments and generates high-quality meshes for specified objects in open-world scenes (top). OMEGAS can effectively reconstruct the unseen portion of the target (bottom).

I. Introduction

RECENT years, the field of 3D reconstruction has emerged as a pivotal area of research, driven by its profound applications across a diverse range of disciplines, including robotics [1], architectural design [2], virtual reality [3], and so on. The community has successfully achieved high-quality, real-time reconstruction and rendering of complex 3D scenes, largely due to the advancement of 3D rendering-based models [4]–[9].

Recently, state-of-the-art methods have not only achieved highly accurate 3D reconstruction of entire scenes but have also begun exploring 3D consistent scene segmentation. SPIn-NeRF [10] introduces a method for segmenting specific objects within NeRF and achieving 3D-consistent inpainting across views. Building on this, Gaussian Grouping [11] extends the object-centric concept to 3D Gaussian Splatting [7], enabling unified reconstruction, segmentation, and versatile editing in open-world 3D scenes.

This work was supported by the National Natural Science Foundation of China (Grant No. 62173045), the Beijing Natural Science Foundation under Grant F2024203115, and BUPT Excellent Ph.D. Students Foundation (CX20241088). (Lizhi Wang and Feng Zhou are co-first authors.) (Corresponding author: Jianqin Yin.)

Lizhi Wang, Feng Zhou, Bo Yu, Pu Cao, and Jianqin Yin are with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: wanglizhi@bupt.edu.cn, zhoufeng@bupt.edu.cn, a7858833@bupt.edu.cn, caopu@bupt.edu.cn, jqyin@bupt.edu.cn).

However, existing methods struggle to reconstruct a specific target object's 3D mesh in high-quality given scene images, which is somehow more solid and straightforward in downstream applications like virtual game modeling. This challenge is particularly evident in two key aspects. First, the reconstruction of entire scenes often leads to a compromise in the quality of specific object reconstruction. Besides, certain parts of a specific object in a scene are frequently occluded or invisible from any perspective, making their reconstruction challenging with current methods, as illustrated in Figure 1 bottom.

In this paper, we propose a novel and effective framework that reconstructs the 3D mesh of the target object given multi-view open-world scene images, termed OMEGAS: Object Mesh Extraction from Large Scenes Guided by GAussian Segmentation. As shown in Figure 1 top, given the multi-view scene images, our framework can freely select and segment the target object from images and extract the complete 3D object mesh.

Specifically, we propose a novel 3D target segmentation technique based on 2D Gaussian Splatting (2DGS) [12]. Inspired by Gaussian Grouping [11], we first utilize the SAM [13] to provide initial target segmentation masks in multi-view scenes. Then, we input the masks along with the scene images into a 2DGS model and introduce a unique compact identity vector as a supervisory signal, enabling 3D

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2025.3586755

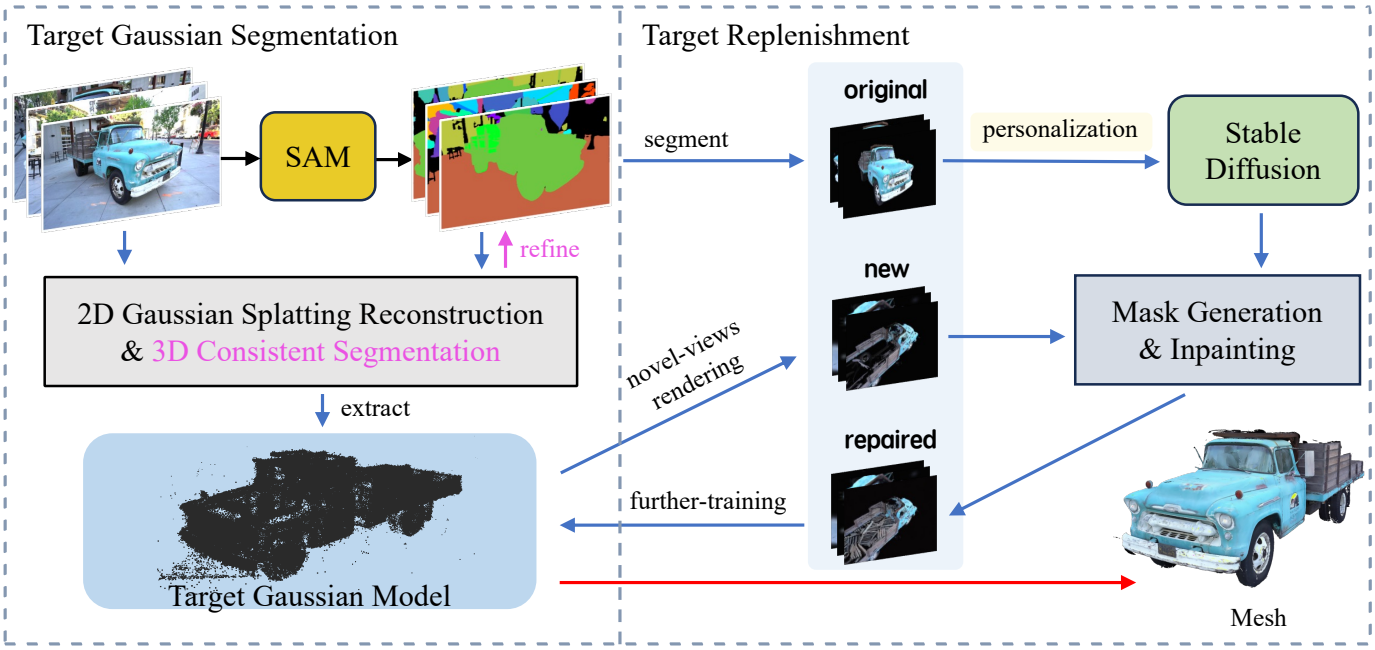


Fig. 2: The framework of **OMEGAS**. OMEGAS comprises two stages: **Target Gaussian Segmentation** to segment target 2DGS model from multi-view images and provide accurate 3D consistent masks; **Target Replenishment** to optimize the target model by personalized Stable Diffusion with a Mask-generation & Inpainting process. The final object mesh is extracted from the optimized target Gaussian model.

segmentation within the 2DGS space. By leveraging the 3D consistency of the 2DGS model, we can iteratively optimize the segmentation results of the target within the scene, ultimately obtaining precise 2D segmentation results along with an initial 3D model of the target.

Subsequently, to address the occluded portions of the target that cannot be directly reconstructed, we leverage large-scale, open-world generative priors such as Stable Diffusion [14]. Specifically, we segment the target from the original perspective and then use these target-only images to personalize Stable Diffusion. This approach allows for the concentration of knowledge on the specific target. Our goal is to utilize the customized Stable Diffusion to assist inpainting the incomplete faces of the target Gaussian model. In this process, we propose a novel mask generation technique based on the diffusion denoising process to mark the under-constructed portions of the novel-view rendered images. Ultimately, we further train the 2DGS model with additional inpainted view images to obtain the final, fully completed target mesh.

Extensive experiments demonstrate the superior performance of our framework, capable of reconstructing the target object mesh with high precision.

In summary, our contributions are:

- We propose OMEGAS, an effective framework to extract meshes of specified objects through multi-view 2D images in open-world scenes.
- We propose a novel 3D target segmentation technique based on 2D Gaussian Splatting to extract the target 3D model from multi-view scene images and create 3D-consistent target masks. Furthermore, we introduce a novel target replenishment technique by leveraging large-scale generative diffusion priors to adaptively optimize the unseen portions of the target.
- Extensive experiments reveal that OMEGAS operates effectively in a range of open-world scenes and significantly surpasses the existing approach in performance. Specifically, our approach shows superiority in both texture details and occlusion robustness on target object reconstruction.

II. Related Work

In this section, we first review several methods for the segmentation of rendering-based 3D models, then briefly review the existing methods for mesh extraction from images. Finally, we review some approaches to 3D reconstruction from partial views.

A. Segmentation of Rendering-based 3D Models

Due to the intrinsic implicit nature of the rendering-based model, it's hard to do normal data-driven training for semantic segmentation like explicit 3D models (e.g., 3D point-cloud segmentation [15]–[23]). To address this issue, SPIn-NeRF [10] first proposes a method that segments specific objects in NeRF and achieves inpainting in different views with 3D consistency. Recently, 3D Gaussian Splatting established its effectiveness in the reconstruction task exhibiting high inference speeds and remarkable quality. Following SPIn-NeRF, Gaussian Grouping [11] extends the object-oriented concept from SPIn-NeRF to 3D Gaussian Splatting, enabling the joint reconstruction and segmentation of anything in open-world 3D scenes and various 3D editing tasks.

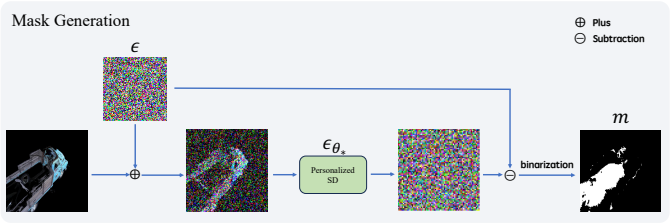


Fig. 3: Mask generation process. We estimate the noise using a personalized diffusion model and calculate the noise residual. Then, we rescale the results and apply a threshold for binarization to determine which parts of the image need to be changed.

B. Mesh Extraction from Images

In the early days, the development of Structure-from-motion (SfM) [24]–[26], Multi-View Stereo (MVS) [27]–[29] and other methods [30]–[34] allows for 3D reconstruction from multi-view images. More recently, owing to the development of neural network-based 3D reconstruction method [4], [7], various approaches have explored integrating rendering-based models with mesh reconstruction [35]–[39]. For example, some works optimize neural signed distance functions (SDF) by training neural radiance fields (NeRF) in which the density is derived as a differentiable transformation of the SDF [37], [38]. A triangle mesh can finally be reconstructed from the SDF by applying the Marching Cubes algorithm [40]. Notably, SuGaR [41] introduced the pioneering method for high-precision scene mesh reconstruction from a 3DGS model. In contrast, our frameworks focus on specific target mesh reconstruction from scenes.

C. 3D Reconstruction from Partial Views

Vanilla methods like NeRF [4] or 3DGS [11] struggle in reconstruction under partial-view settings. Some research efforts focused on incorporating auxiliary priors, such as depth maps [42], [43], information theory [44], symmetry [45], and continuity [46] to infer missing information. However, these approaches tend to be either too simplistic or overly specialized for certain scenes, resulting in poor generalization.

The recent progress in text-to-image diffusion models [14] and their tailored applications [47], [48] in the 3D field make it possible for reasonable novel-view synthesis, paving the way for impactful applications such as single-view 3D asserts generation (image-to-3d) [49], [50] and sparse reconstruction [51]–[53]. However, existing sparse reconstruction methods are designed for scenarios where camera views are sparsely distributed across a 360 range. In contrast, our approach aims to reconstruct partially occluded targets by focusing on cases where views are densely clustered within the visible range of the target, while the occluded views remain entirely absent.

III. Method

Our framework aims to reconstruct the precise mesh of the target object from multi-view scene images. It comprises two main components: Target Gaussian Segmentation (TGS) to segment the 3D target from multi-view scene images and Target Replenishment (TR) to optimize the target by large-scale diffusion priors, as illustrated in Figure 2.

The rest of this section is organized as follows: in section Preliminaries, we introduce some basic concepts involved in this section, including 2D Gaussian Splatting [12] and the generative priors. In section Target Gaussian Segmention, we introduce the technique that segments the initial 3D target from multi-view images by 2D Gaussian Splatting. In section Target Replenishment, we introduce the process that adaptively optimizes the target model by diffusion priors.

A. Preliminaries

2D Gaussian Splatting (2DGS) [12]. Due to the multi-view inconsistent nature of 3D Gaussians, 3D Gaussian Splatting (3DGS) fails to accurately represent surfaces. Thus, [12] proposed 2D Gaussian Splatting, which collapses the 3D volume into a set of 2D oriented planar Gaussian disks and provides view-consistent geometry while modeling surfaces intrinsically. Specifically, a 2D Gaussian is defined in a local tangent plane in world space, which is parameterized:

$$P(u,v) = \mathbf{p}_k + s_u\mathbf{t}_u u + s_v\mathbf{t}_v v = \mathbf{H}\left(u,v,1,1\right)^\top \quad (1)$$

$$\text{where } \mathbf{H} = \begin{bmatrix} s_u\mathbf{t}_u & s_v\mathbf{t}_v & 0 & \mathbf{p}_k \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{RS} & \mathbf{p}_k \\ 0 & 1 \end{bmatrix} \quad (2)$$

where a 2D Gaussian plane $P(u,v)$ is defined by its central point $\mathbf{p}_k$, principal tangential vectors t_u and t_v, scaling vectors s_u and s_v. $\mathbf{H} \in 4\times 4$ is the homogeneous transformation matrix representing the geometry of the 2D Gaussian. For the point $\mathbf{u} = (u,v)$ in uv space, its 2D Gaussian value is evaluated by standard Gaussian:

$$\mathbf{G}(\mathbf{u}) = \exp\left(-\frac{u^2+v^2}{2}\right) \quad (3)$$

The center $\mathbf{p}_k$, scaling (s_u, s_v), and the rotation $(\mathbf{t}_u, \mathbf{t}_v)$ are learnable parameters. Our framework harnesses the 2DGS model as the target reconstruction carrier due to its meticulous object reconstruction capabilities.

Text-to-Image Generative Priors & Personalization. In the past few years, with the advent of diffusion-based generative techniques, the community has developed numerous mature open-world large-scale generative models, e.g., Stable Diffusion [14]. Technically, given input image $x \in \mathbf{R}^{H\times W\times 3}$, an encoder of variational auto-encoder (VAE) $\mathcal{E}$ converts it into a latent representation $z_0 \in \mathbf{R}^{h\times w\times c}$, where image is downsampled by a factor f and channels are increased to c. Then a text-conditioned UNet ϵ_θ in latent space is used to predict the noise of each timestamp t from z_t and prompt y to recover z_0, where θ is the param of UNet. After the denoising process, VAE's decoder $\mathcal{D}$ transforms z_0 to image space to generate the image.

The personalization of diffusion priors [54], [55] is a technique that concentrates prior knowledge on a specific object by transforming or fine-tuning UNet parameters θ into a form specific to the given few object images, represented as ϵ_{θ^*}.

Fig. 4: Results of meshes with OMEGAS on the Instruct-NeRF2NeRF dataset(line 1), Tanks&Temples dataset(line 2), LERF dataset(line 3,4,5), Mip-NeRF360 dataset(line 6) and 3D-OVS dataset(line 7). The scenes we selected for validation are complex in both indoor and outdoor settings and contain a large number of objects rather than just a single target. We also show the result of getting multiple objects located in a large scene(line 3,4,7).

B. Target Gaussian Segmentation (TGS)

As depicted in the left part of Figure 2, we begin by introducing a novel technique known as Target Gaussian Segmentation. This method segments 3D-consistent target masks from multi-view images and extracts a preliminary 2DGS model of the target. The process is as follows.

Scene Segmentation by SAM. We first conduct preliminary target consistency segmentation on scene images from different views based on SAM's capabilities in open-world segmentation for intricate scenes. Specifically, following the approach in [11], we treat the scene images from divergent views as a continuous video sequence and input them into SAM for initial segmentation. Next, by applying a zero-shot tracker [56] to these segmentation results, we obtain unique object IDs ranging from 0 to 255. The segmented outcomes are preserved as gray-scale images, with each object's ID represented in corresponding gray-scale values.

Target 2DGS Reconstruction & 3D Consistent Segmentation. We leverage the gray-scale images to guide the segmentation of 2DGS. We treat the gray scale as attributes similar to RGB colors for training 2D Gaussians. Similar to representing Gaussian colors using spherical harmonics coefficients, we add identity vectors to 2D Gaussians following [10], [11]. To reduce the memory usage and training time, we set the identity vectors as length 8, encoding segmentation labels ranging from 0 to 255. When a 2D Gaussian is observed from a slanted

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2025.3586755

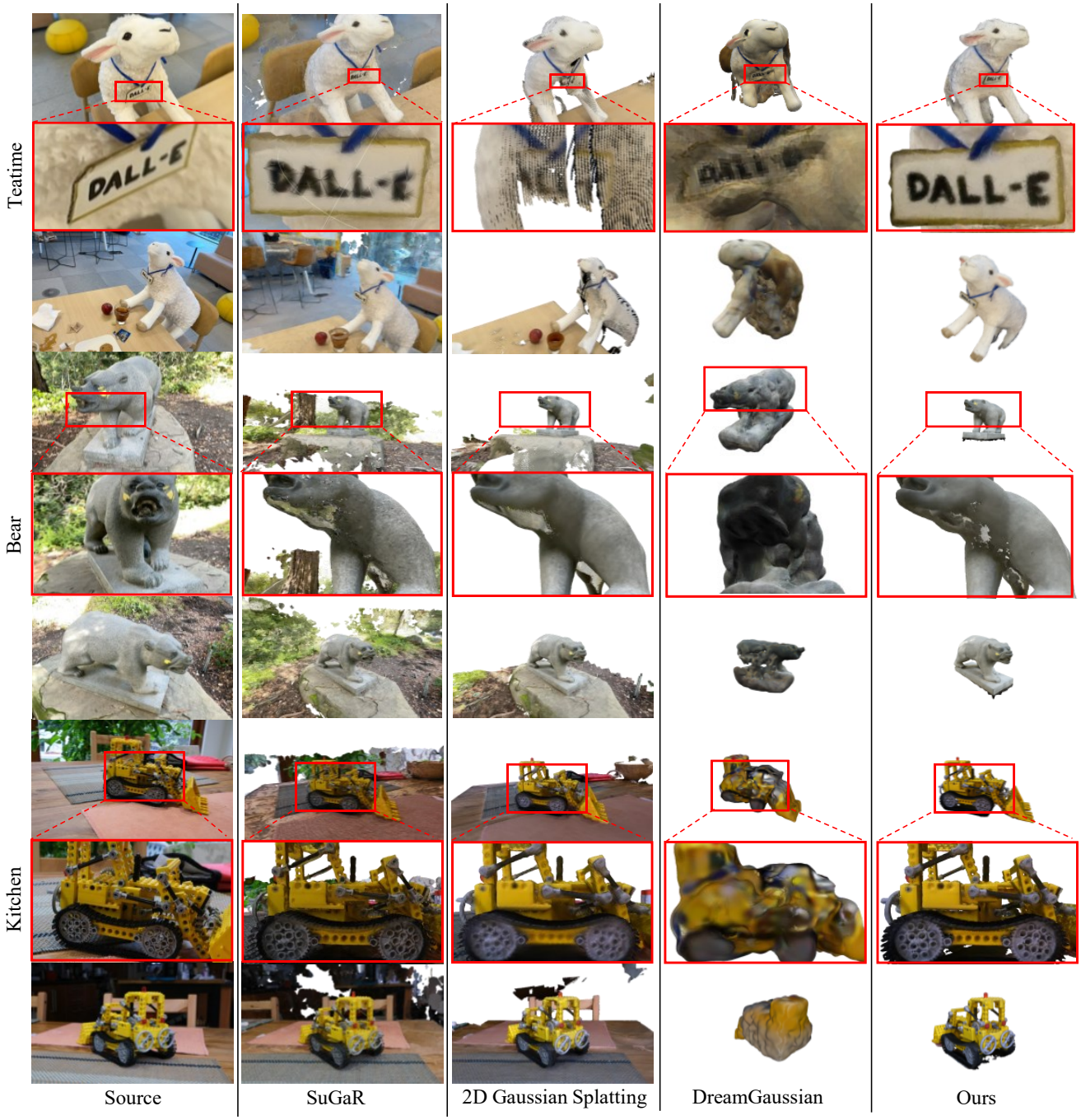


Fig. 5: Comparing mesh extracting results with SuGaR, 2D Gaussian Splatting and DreamGaussain. The scenes are selected from LERF, Instruct-NeRF2NeRF and Mip-NeRF360 datasets. Both SuGaR, 2DGS, and our approach use multi-view images as input, while DreamGaussian relies on a single-view image.

views, the object-space low-pass filter is utilized:

$$\hat{\mathbf{G}}(\mathbf{x}) = \max\left\{\mathbf{G}(\mathbf{u}(\mathbf{x})), \mathbf{G}(\frac{\mathbf{x}-\mathbf{c}}{\sigma})\right\} \tag{4}$$

where $\mathbf{c}$ is the projection of center $\mathbf{p}_k$.

By conducting differentiable rendering of identity vectors, similar to rendering colors, by blending $\mathcal{N}$ ordered Gaussians on overlapping pixels, we can calculate the identity vectors O of pixels:

$$O(\mathbf{x}) = \sum_{i\in\mathcal{N}} o_i \alpha_i' \hat{\mathbf{G}}_i(\mathbf{u}(\mathbf{x})) \prod_{j=1}^{i-1}(1-\alpha_j'\hat{\mathbf{G}}_j(\mathbf{u}(\mathbf{x}))) \tag{5}$$

where o_i represents the identity vector of each Gaussian, and α_i' is given by evaluating the opacity of 2D Gaussians multiplied by the opacity of each point. Unlike colors, the identity vectors of the same object does not change with different viewpoints, so we set the SH degree to 0, reduc-

ing computational complexity. Since the identity vectors are obtained through rendering, they differ from the IDs generated by SAM and exhibit 3D consistency, ensuring greater stability across varying perspectives.

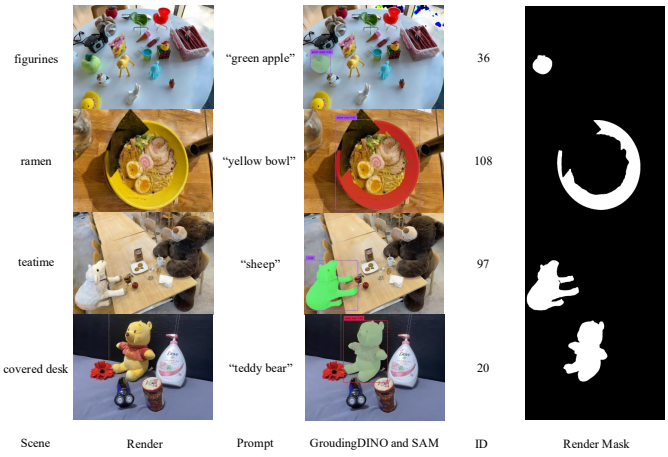


Fig. 6: Visualization of target gaussian segmentation on LERF-MASK dataset [11] and 3D-OVS dataset [57]

Segment Loss. Next, we use the 3D consistency of 2DGS to optimize the consistency of segmentation results on the scene images. The L1 loss and SSIM loss in the original 2DGS would lead to the inability to fit object labels. To address this issue, we introduce classification loss and 3D cosine similarity loss. Specifically: 1) for classification loss, we input the rendered identity vectors O into a linear layer$f$ followed by a softmax operation:

$$F(O) = \text{softmax}(f(O)) \tag{6}$$

Then we use the standard cross-entropy loss L_{oe} for classification; 2) For the 3D cosine similarity loss, we sample m 2D Gaussians, ensuring that the cosine similarity of the identity features F_o from the n nearest 2D Gaussians is closely aligned:

$$L_{cs} = \frac{1}{mn}\sum_{j=1}^{m}\sum_{i=1}^{n}\frac{F(o_j)\cdot F(o_i)}{\|F(o_j)\|\|F(o_i)\|}$$

Making similar identity vectors closer can improve the 3D consistency of segmentation, thus enhancing segmentation accuracy. Following 2D Gaussian Splatting, we construct the 2D Gaussian loss function L_{gs} using RGB, depth, and normal:

$$L_d = \sum_{i,j}\omega_i\omega_j|z_i - z_j| \tag{7}$$

where $\omega_i = \alpha_i \hat{G}_i(u(x))\prod_{j=1}^{i-1}(1-\alpha_j \hat{G}_j(u(x)))$ is the blending weight of the $i-$th intersection and z_i represents the depth of the intersection points. When calculating depth, unlike the 2D Gaussian Splatting [12] which defaults to using the average depth, we utilize the proposed median depth as the largest visible depth, considering $T_i = 0.5$ as the pivot for surface and free space:

$$z_{\text{median}} = \max\{z_i | T_i > 0.5\}. \tag{8}$$

Subsequently, we align the normals of the splats with the gradients of the depth maps as follows:

$$L_n = \sum_i \omega_i(1 - n_i^T N) \tag{9}$$

where i indexes the intersected splats along the ray, ω denotes the blending weight of the intersection point, n_i represents the normal of the splat facing the camera, and N s the normal estimated from the gradient of the depth map. Specifically, N is computed using finite differences from nearby depth points as follows:

$$N(x,y) = \frac{\nabla_x p_s \times \nabla_y p_s}{|\nabla_x p_s \times \nabla_y p_s|} \tag{10}$$

The 2D Gaussian loss function L_{gs} is as follows:

$$L_{gs} = L_c + \alpha L_d + \beta L_n \tag{11}$$

where L_c is an RGB reconstruction loss combining L_1 with the D-SSIM term from [7], while L_d and L_n are regularization terms.

The total loss function is the weighted sum of the segmentation loss function and the 2D Gaussian loss function L_{gs}:

$$L = L_{gs} + \lambda_{oe}L_{oe} + \lambda_{cs}L_{cs} \tag{12}$$

When SAM produces masks with incorrect shape or identity encoding, TGS can learn unique identity encodings through cross-entropy loss and 3D cosine similarity loss.

Target Extraction. After training the 2DGS model to convergence, we extract the target model using its object ID. For extraction, we adopt Grounding DINO [58] and SAM [59] to segment a scene image, obtaining a mask corresponding to the text prompt. We select the IDs within the mask area that exceed a threshold of p_{id} as the IDs for the corresponding objects. Subsequently, we utilize a classifier to obtain the category probabilities for each Gaussian. We then select the Gaussians that have a probability greater than p_{ex} for belonging to this ID, as well as the Gaussians that are encompassed within its convex hull, for extraction and preservation. As mentioned in implementation details of our paper, the threshold p_{ex} for extracting Gaussians is set to 0.95. Meanwhile, the threshold p_{id} for extraction is 0.5. By rendering this model back into the original views, we obtain precise 3D-consistent target masks.

C. Target Replenishment (TR)

As shown in the right section of Figure 2, after segmenting the target objects, we extract the target 2DGS model and obtain precise target masks in the view images. We then utilize generative diffusion priors to adaptively replenish the unseen portions and further refine the target model. In summary, we render the unseen view of the target Gaussian model, inpaint the areas that are not sufficiently reconstructed by Stable Diffusion, and then feed the inpainted images for further training.

Stable Diffusion Personalization. To focus the knowledge of large-scale diffusion priors on a specific target, we employ a personalization method to control Stable Diffusion. For each object, we fine-tune a personalized diffusion model to optimize

TABLE I: Comparison of segmentation on LERF-MASK dataset [11] and 3D-OVS dataset [57]. For segmentation results, we followed standard practices from classic 3D scene segmentation methods [11], [60], computing the mIOU and mBIoU between predicted segmentation masks and ground truth across multiple views. We also compared the average memory required for training. The empirical evidence substantiates that our methodology yields superior outcomes in both quality and efficiency.

Model	Mem	figurines			ramen			teatime			covered desk		
	GB ↓	mIoU↑	mBIoU↑	time↓	mIoU↑	mBIoU↑	time↓	mIoU↑	mBIoU↑	time↓	mIoU↑	mBIoU↑	time↓
SAM		74.83	73.71		57.25	56.99		75.95	74.48		75.23	74.17	
GaussianGrouping (7k)	21	86.08	84.08	7m	70.34	58.03	7m	75.98	71.91	7m	75.28	72.45	7m
Ours (7k)	**10**	**86.21**	**84.09**	**6m**	**86.48**	**73.62**	**5m**	**78.81**	**73.32**	**6m**	**80.74**	**77.98**	**5m**
Ours (30k)	**13**	**88.86**	**86.65**	**65m**	**88.63**	**75.93**	**51m**	**79.41**	**73.37**	**64m**	**81.48**	**79.29**	**55m**

TABLE II: Prompt labels used during segmentation experiments on LERF-Mask dataset [11] and 3D-OVS dataset [57].

Scene	Text queries	
ramen	chopsticks	egg
	pork belly	yellow bowl
figurines	green apple	green toy chair
	old camera	porcelain hand
	red toy chair	rubber duck with red hat
teatime	bag of cookies	cookies on a plate
	sheep	apple
	paper napkin	coffee mug
	bear	tea in a glass
covered desk	teddy bear	flower
	shaver	shampoo

the object. Specifically, we use the target Gaussian model to render the masks of the target object, thereby obtaining target images$\tilde{I}$ of the target object from the original views. Then we input the target images into a personalization model $\mathcal{P}$ (e.g., DreamBooth [54]).

$$\epsilon_{\theta^*} = \mathcal{P}(\tilde{I}, \epsilon_\theta) \tag{13}$$

where ϵ_θ is the UNet of original Stable Diffusion with parameter θ, and ϵ_{θ^*} is the UNet of personalized SD.

Mask Generation & Inpainting. To determine the regions that require repair, We then use the personalized SD to generate the poorly reconstructed mask of the novel-view rendering of the target Gaussian model, as depicted in Figure 3. The mask is identified as the differences between the complete target object and the target object with holes. Specifically, given a novel-view image x_0, we encode it into latent space by the VAE encoder $\mathcal{E}$.

$$z_0 = \mathcal{E}(x_0) \tag{14}$$

For the novel view image x_0, we render by randomly generating n camera poses. For most datasets, we set n to 100. For objects with significant missing parts, we utilize PyTorch3D [61] to render the object's mesh. Compared to using 2D gaussian of the object, we find that mesh rendering reveals more pronounced gaps, resulting in more accurate masks.

We then perturb z_0 by diffusion forward process with noise ϵ:

$$z_t = \sqrt{\overline{\alpha}_t} z_0 + \sqrt{1 - \overline{\alpha}_t}\, \epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}),$$

where $\mathcal{N}(0, \mathbf{I})$ is a Gaussian Distribution, t denotes the timestep sampled from $[0, T]$, z_t represents perturbed z_0 at t, and $\overline{\alpha}_t$ is predefined noise scheduling coefficient.

By applying the personalized UNet ϵ_{θ^*}, we can get the noise residual by:

$$\triangle\epsilon = \epsilon_{\theta^*}(z_t, y, t) - \epsilon,$$

where $\triangle\epsilon$ is the latent noise residual, y represents the embedded text prompt, which we set to empty based on experimental results.

The final inpainting mask m is given by:

$$m = \mathcal{B} \circ \mathcal{D}(\triangle\epsilon),$$

where $\mathcal{D}$ is the VAE decoder, and $\mathcal{B}$ is the image binarization operator.

Subsequently, the mask m and the novel-view image x_0 are input into a standard Stable Diffusion inpainting model[1], resulting in the final fixed image $\widetilde{x}_0$.

Mesh Extraction. By generating n novel-view images $X = \{x_0^i\}_{i=0}^{n-1}$ and applying the inpainting process, we obtain n refined images $\widetilde{X} = \{\widetilde{x}_0^i\}_{i=0}^{n-1}$. These refined images are then fed into the target 2DGS model for further training, addressing deficiencies in the first stage.

Ultimately, following [12], to obtain meshes from the reconstructed 2D splats, we generate depth maps of the training views by projecting the splats' depth values onto the pixels. We then apply truncated signed distance fusion (TSDF) using Open3D to merge the reconstructed depth maps.

IV. Experiment

In this section, we present our experimental settings and extensive results with in-depth analysis. Firstly, we carry out ablation studies for the proposed method on the DroneVehicle dataset and then discuss the operations that affect the performance of the proposed method. Finally, the mAP of our method and the state-of-the-arts are reported, and the visualization of the detection results is discussed.

A. Implementation Details

All experiments are performed and measured on a single GPU NVIDIA RTX 3090 with Ubuntu 18.04 LTS.

[1]huggingface.co/stabilityai/stable-diffusion-2-inpainting

TABLE III: Quantitative results of target mesh extraction on the Tanks & Temples dataset [62], represented by F1 score and time.

Scene	SuGaR	DreamGaussian	2DGS	**Ours**
Truck	0.0741	0.0013	0.1421	**0.1635**
Ignatius	0.1392	0.0007	0.3423	**0.4464**
Caterpillar	0.1827	0.0031	0.2100	**0.2208**
Mean	0.1320	0.0017	0.2315	**0.2769**
Time	1 h	3 m	16 m	20 m

Datasets. To evaluate the reconstruction quality, we tested our OMEGAS on scenes presented in LERF-MASK dataset [11] and Mip-NeRF360 dataset [5], where the flowers and treehill are skipped due to the non-public access right. We also take diverse 3D scene cases from LERF [63], Tanks&Temples [62], 3D-OVS [57] and Instruct-NeRF2NeRF [64] for visual comparison.

Model Details. In Target Gaussian Segmentation, we take SAM-HQ model [59] for initial segmentation The confidence threshold p_{ex} for extraction is 0.95. We use the Adam [65] optimizer for both Gaussians and linear and train for max 30000 iterations with a learning rate of 0.0025 for identity vectors and 0.0005 for linear layer. For 3D regularization loss, we choose $n = 5$ and $m = 1000$. We start the densification process from the 500th iteration, with a densification interval set to 100. Specifically, we extend the densification process until the 30000th iteration of training to continuously eliminate erroneous gaussians generated by the normal vector loss. The interval for resetting opacity is set to 3000. Every 5 iterations, we use the sum of the segmentation loss and gaussian loss as the loss function, while for the rest, we only use the 2D gaussian loss. We employ principal component analysis as a means to visualize the outcomes of the segmentation.

In Target Replenishment, we adopt Stable Diffusion 2.1 with the resolution 512x512. We take DreamBooth [54] as our personalization model. When generating a mask, we fix the timestep t to 991 and sample 10 random seeds to obtain an averaged mask result.

For mask generation, we apply erosion and dilation operations to the masks generated by stable diffusion to achieve smoother mask edges, using a kernel size of 5x5. For future training, unlike target gaussian segmentation, we enabled depth loss and disabled segmentation loss, setting α to 1000.

For mesh extraction, we set the truncated threshold to 0.02 and the voxel size to 0.004 during TSDF fusion. For most datasets, we set the depth truncation to 4.0.

The parameters of our method are consistent across datasets, with only the prompts and mesh extraction parameters varying for different objects. We use a fixed random seed (seed = 64) to ensure reproducibility and consistency across ablation experiments. For comparative evaluations, we employ random seeds to assess the robustness and generalizability of our method.

B. Comparative Analysis

Target Mesh Extraction. To demonstrate the effectiveness of our target mesh extraction method, we provide qualitative comparisons with the scene reconstruction approach SuGaR [41] and 2DGS [12]. Since our method is the first to tackle target reconstruction in a multi-view context, we are limited to using the single-view method DreamGaussian [49] as our other target reconstruction comparative baseline. The results are presented in Figure 5, showcasing detailed qualitative outcomes across various open-world scenarios, including teatime, bear, and kitchen from the LERF [63], Instruct-NeRF2NeRF [64] and Mip-NeRF360 [66] datasets. Both SuGaR, 2DGS, and our approach use multi-view images as input, while DreamGaussian relies on a single-view image.

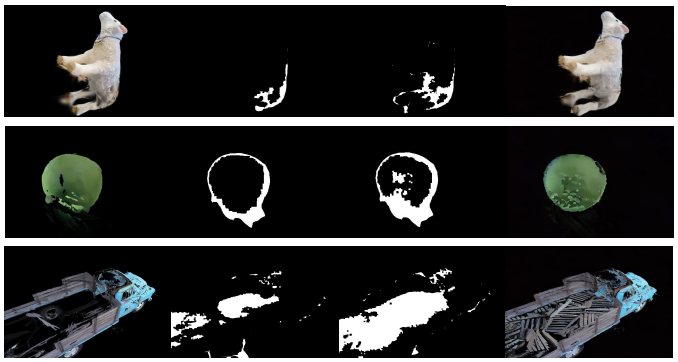
Compared to SuGaR and 2DGS, our method achieves superior detail across all three scenes. SuGaR and 2DGS's focus on the overall scene lead to a noticeable reduction in the granularity of local details. In the bear scene, our method demonstrates its capability to reconstruct unseen portions in the input view. Meanwhile, the kitchen scene highlights the precision of our approach in segmenting and extracting complex objects. We believe that the poor performance of SuGaR is due to the large scene size, as SuGaR attempts to reconstruct the entire scene, leading to the loss of some object details. Our method, on the other hand, reconstructs the target object through segmentation, allowing for the use of a higher number of meshes and shorter reconstruction time for a single object under the same memory constraints. Additionally, our method addresses the issue of holes caused by insufficient input views, which SuGaR does not optimize for. Compared to DreamGaussian, our method significantly enhances the quality of the generated meshes. Although DreamGaussian can generate the missing parts of the target object, it does not perform 3D reconstruction based on input views, and the diffusion model generates images with lower resolution, resulting in poorer surface details.

We also conduct quantitative experiments in Table III on the Tanks & Temples dataset [62]. Since our method focuses on mesh extraction of target objects within a scene, we manually removed other objects from the ground truth mesh to quantify the accuracy of our approach. For methods that cannot isolate the target object, we used the complete ground truth mesh to ensure fairness. We employed the evaluation metrics provided by the Tanks&Temples dataset [62], calculating precision and recall to obtain the F1 score. Our method consistently achieves superior results across all scenarios in the dataset compared to other methods.

In Figure 4, we further demonstrate the mesh quality and highlight our method's effectiveness in extracting small objects from complex scenes.

Scene Segmentation. To demonstrate the effectiveness of our Gaussian Segmentation method, we conducted quantitative scene segmentation experiments on the LERF-MASK dataset [63]based on the LERF-Localization evaluation dataset [63], using Gaussian Grouping [11], as shown in Table I. For each 3D scene, we use 6 text queries with corresponding GT mask label in average. Similar to the annotation used in LERF

Fig. 7: Effect of the Target Replenishment (TR) Module. The first line shows the mesh generated solely using Target Gaussian Segmentation. The second line displays the mesh produced by our complete model, including the Target Replenishment module. The significant loss of information in the input views results in numerous missing parts in the mesh. However, experimental results show that TR can fill in the missing parts while keeping the visible parts unchanged.



(a) Novel View Image (b) Mask w/o Personalization (c) Mask w/ Personalization (d) Repaired Image

Fig. 8: Effect of personalization and inpainting. It shows the mask w/ or w/o personalization of SD and the inpainting results on the Tanks&Temples dataset and LERF dataset. The mask created by the personalized Stable Diffusion model spans a larger area and more precisely pinpoints the locations of the holes, leading to a superior repaired image.

[63], for each of the 3 scenes, we choose 2-4 novel views for testing and annotating the rendering of novel views. All language prompts used for LERF-Mask dataset evaluation are listed in Table II, which contains 18 prompts in total. Due to VRAM limitations, we trained Gaussian Grouping for a max of 7,000 iterations on a single NVIDIA RTX 3090 GPU. Despite these constraints, our method achieved considerable improvements under the same iteration settings, with lower memory usage and reduced training time across three scenes. When further training our method for 30,000 iterations, we observed more performance gains.

We provide visualization on target gaussian segmentation in Figure 6. We use four scenes from the LERF dataset [63] and 3D-OVS dataset [57], each scene being input with an object prompt. Through Gaussian segmentation, we obtain the ID and 2DGS of the corresponding object. Our presentation encompasses the images rendered from each scene, the segmentation outcomes derived from the utilization of Grounding DINO [58] and SAM [59], along with the masks rendered from the 2D gaussian of the extracted objects.

TABLE IV: Effect of Personalizing SD on the Tanks & Temples dataset [62] and LERF-MASK dataset [11]. It shows the accuracy of mask generated by personalized SD is higher than that of SD, thus enabling more accurate filling of unseen portions.

Model	Truck	Teatime	Figurines	Mean
SD	0.3434	0.2317	0.2826	0.2859
Personalized SD	0.6543	0.5842	0.5428	0.5938

C. Ablations

In this section, we isolate the design choices and evaluate their impacts, including the effect of the Target Replenishment stage and the necessity of personalizing SD.

Effect of Target Replenishment (TR). We first examine the effects of the proposed Target Replenishment module, as shown in Figure 7. It shows the difference between mesh results w/ and w/o TR module on an example of Tanks&Temples dataset [62]. Specifically, there are no images taken from above or showing the interior of the truck's carriage in the given dataset views. Without TR, a significant mesh hole would appear inside the truck, whereas our method can properly fill the mesh. The significant loss of information in the input views results in numerous missing parts in the mesh. However, experimental results show that TR can fill in the missing parts while keeping the visible parts unchanged.

Effect of Personalizing SD. We now analyze the effect of personalizing SD to generate inpainting masks in Figure 8. In the figure, (a) are images of the object rendered from a random

viewpoint, (b) are masks generated by a standard Stable Diffusion model, (c) are masks generated by a personalized Stable Diffusion model, and (d) are inpainted images by mask (c). It can be observed that the mask generated by the personalized Stable Diffusion model covers a larger area and more accurately identifies the locations of the holes, resulting in a better-repaired image. To further illustrate its effectiveness, we conducted a quantitative experiment in Table IV: we selected several classic target meshes and manually removed one of their faces. We then compared the inpainting mask coverage generated by SD with and without personalization. The results clearly highlight the advantages of personalization.

V. Limitations

For Target Gaussian Segmentation (TGS), our method relies on SAM's initial segmentation and 2DGS for scene reconstruction, making it challenging to accurately segment and reconstruct extremely small objects. For Target Replenishment (TR), errors in diffusion model generation may lead to reduced performance. Since 2DGS lacks the capability to reconstruct dynamic scenes, our method cannot handle moving objects. Future advancements in segmentation models could further enhance the performance of our approach. Applying our method to approaches like 4DGS could potentially enable dynamic object segmentation and optimization.

VI. Conclusion

We present OMEGAS: Object Mesh Extraction from Large Scenes Guided by Gaussian Segmentation. OMEGAS efficiently extracts high-precision meshes of target objects from multi-view scene images and can reconstruct occluded or invisible parts of the targets. OMEGAS introduces a novel Target Gaussian Segmentation technique, which segments the target Gaussian model from multi-view images using 2D Gaussian Splatting. Additionally, OMEGAS proposes a Target Replenishment technique that leverages open-world diffusion priors to address unseen portions of the target.

References

[1] B. Siciliano, O. Khatib, and T. Kröger, *Springer handbook of robotics*. Springer, 2008, vol. 200.

[2] I. Caetano, L. Santos, and A. Leitão, "Computational design in architecture: Defining parametric, generative, and algorithmic design," *Frontiers of Architectural Research*, vol. 9, no. 2, pp. 287–300, 2020.

[3] J. Xiong, E.-L. Hsiang, Z. He, T. Zhan, and S.-T. Wu, "Augmented reality and virtual reality displays: emerging technologies and future perspectives," *Light: Science & Applications*, vol. 10, no. 1, pp. 1–30, 2021.

[4] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020.

[5] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-nerf 360: Unbounded anti-aliased neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5470–5479.

[6] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelnerf: Neural radiance fields from one or few images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4578–4587.

[7] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, pp. 1–14, 2023.

[8] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, "4d gaussian splatting for real-time dynamic scene rendering," *arXiv preprint arXiv:2310.08528*, 2023.

[9] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, and X. Jin, "Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction," *arXiv preprint arXiv:2309.13101*, 2023.

[10] A. Mirzaei, T. Aumentado-Armstrong, K. G. Derpanis, J. Kelly, M. A. Brubaker, I. Gilitschenski, and A. Levinshtein, "Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields," in *CVPR*, 2023.

[11] M. Ye, M. Danelljan, F. Yu, and L. Ke, "Gaussian grouping: Segment and edit anything in 3d scenes," *ArXiv*, vol. abs/2312.00732, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:265551523

[12] B. Huang, Z. Yu, A. Chen, A. Geiger, and S. Gao, "2d gaussian splatting for geometrically accurate radiance fields," in *ACM SIGGRAPH 2024 Conference Papers*, 2024, pp. 1–11.

[13] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *ICCV*, 2023.

[14] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[15] Y. Liu, Q. Hu, and Y. Guo, "Bsts: A weakly-supervised method for semantic learning of 3d point clouds," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 11, pp. 11 386–11 399, 2024.

[16] C. Xu, B. Wu, Z. Wang, W. Zhan, P. Vajda, K. Keutzer, and M. Tomizuka, "Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*. Springer, 2020, pp. 1–19.

[17] S. Fan, Q. Dong, F. Zhu, Y. Lv, P. Ye, and F.-Y. Wang, "Scf-net: Learning spatial contextual features for large-scale point cloud segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 504–14 513.

[18] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3d point clouds: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 12, pp. 4338–4364, 2021.

[19] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[20] W. Wu, Z. Qi, and L. Fuxin, "Pointconv: Deep convolutional networks on 3d point clouds," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2019, pp. 9621–9630.

[21] Y. Wang, L. Wang, Q. Hu, Y. Liu, Y. Zhang, and Y. Guo, "Panoptic segmentation of 3d point clouds with gaussian mixture model in outdoor scenes," *Visual Intelligence*, vol. 2, no. 1, p. 10, 2024.

[22] Q. Hu, B. Yang, S. Khalid, W. Xiao, N. Trigoni, and A. Markham, "Towards semantic segmentation of urban-scale 3d point clouds: A dataset, benchmarks and challenges," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 4977–4987.

[23] F. Yin, Z. Huang, T. Chen, G. Luo, G. Yu, and B. Fu, "Dcnet: Large-scale point cloud semantic segmentation with discriminative and efficient feature aggregation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 8, pp. 4083–4095, 2023.

[24] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: exploring photo collections in 3d," in *ACM siggraph 2006 papers*. Association for Computing Machinery, 2006, pp. 835–846.

[25] H. Yin and H. Yu, "Incremental sfm 3d reconstruction based on monocular," in *2020 13th International Symposium on Computational Intelligence and Design (ISCID)*, 2020, pp. 17–21.

[26] R. Shah, A. Deshpande, and P. J. Narayanan, "Multistage sfm: A coarse-to-fine approach for 3d reconstruction," *ArXiv*, vol. abs/1512.06235, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:14620212

[27] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz, "Multi-view stereo for community photo collections," in *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

[28] A. Romanoni, M. Ciccone, F. Visin, and M. Matteucci, "Multi-view stereo with single-view semantic mesh refinement," 2017. [Online]. Available: https://arxiv.org/abs/1708.04907

[29] A. Bignoli, A. Romanoni, and M. Matteucci, "Multi-view stereo 3d edge reconstruction," 2018. [Online]. Available: https://arxiv.org/abs/1801.05606

[30] J. Liu, Y. Hu, J. Yang, Y. Chen, H. Shu, L. Luo, Q. Feng, Z. Gui, and G. Coatrieux, "3d feature constrained reconstruction for low-dose ct imaging," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 5, pp. 1232–1247, 2018.

[31] S. Li, X. Tao, Y. Li, and J. Lu, "Large-scale structured sparse image reconstruction with correlated multiple-measurement vectors using bayesian learning," in *2015 Picture Coding Symposium (PCS)*, 2015, pp. 272–276.

[32] Y. Cheng, Q. Li, R. Li, T. Wang, J. Zhao, Q. Yan, Z. U. Rehman, L. Wang, and Y. Geng, "Lir-net:learnable iterative reconstruction network for fan beam ct sparse-view reconstruction," *IEEE Transactions on Computational Imaging*, vol. 10, pp. 181–195, 2024.

[33] D. S. Alexiadis, A. Chatzitofis, N. Zioulis, O. Zoidi, G. Louizis, D. Zarpalas, and P. Daras, "An integrated platform for live 3d human reconstruction and motion capturing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 4, pp. 798–813, 2017.

[34] Z. Chen, Y. Wang, T. Guan, L. Xu, and W. Liu, "Transformer-based 3d face reconstruction with end-to-end shape-preserved domain transfer," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 12, pp. 8383–8393, 2022.

[35] B. Yang, C. Bao, J. Zeng, H. Bao, Y. Zhang, Z. Cui, and G. Zhang, "Neumesh: Learning disentangled neural mesh-based implicit field for geometry and texture editing," in *European Conference on Computer Vision*. Springer, 2022, pp. 597–614.

[36] F. Darmon, B. Bascle, J.-C. Devaux, P. Monasse, and M. Aubry, "Improving neural implicit surfaces geometry with patch warping," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6260–6269.

[37] Z. Li, T. Müller, A. Evans, R. H. Taylor, M. Unberath, M.-Y. Liu, and C.-H. Lin, "Neuralangelo: High-fidelity neural surface reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8456–8465.

[38] M. Oechsle, S. Peng, and A. Geiger, "Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5589–5599.

[39] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," *arXiv preprint arXiv:2106.10689*, 2021.

[40] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987. [Online]. Available: https://api.semanticscholar.org/CorpusID:15545924

[41] A. Gu'edon and V. Lepetit, "Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering," *ArXiv*, vol. abs/2311.12775, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:265308825

[42] J. Song, S. Park, H. An, S. Cho, M.-S. Kwak, S. Cho, and S. Kim, "Därf: boosting radiance fields from sparse inputs with monocular depth adaptation," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023, pp. 68 458–68 470.

[43] G. Wang, Z. Chen, C. C. Loy, and Z. Liu, "Sparsenerf: Distilling depth ranking for few-shot novel view synthesis," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 9065–9076.

[44] M. Kim, S. Seo, and B. Han, "Infonerf: Ray entropy minimization for few-shot neural volume rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 912–12 921.

[45] S. Seo, Y. Chang, and N. Kwak, "Flipnerf: Flipped reflection rays for few-shot novel view synthesis," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 22 883–22 893.

[46] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. Sajjadi, A. Geiger, and N. Radwan, "Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5480–5490.

[47] Z. Wang, C. Lu, Y. Wang, F. Bao, C. Li, H. Su, and J. Zhu, "Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[48] C. Chen, X. Yang, F. Yang, C. Feng, Z. Fu, C.-S. Foo, G. Lin, and F. Liu, "Sculpt3d: Multi-view consistent text-to-3d generation with sparse 3d prior," *arXiv preprint arXiv:2403.09140*, 2024.

[49] J. Tang, J. Ren, H. Zhou, Z. Liu, and G. Zeng, "Dreamgaussian: Generative gaussian splatting for efficient 3d content creation," *arXiv preprint arXiv:2309.16653*, 2023.

[50] H. M. R. Afzal, S. Luo, and M. K. Afzal, "Reconstruction of 3d facial image using a single 2d image," in *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 2018, pp. 1–5.

[51] X. Liu, J. Chen, S.-h. Kao, Y.-W. Tai, and C.-K. Tang, "Deceptive-nerf: Enhancing nerf reconstruction using pseudo-observations from diffusion models," *arXiv preprint arXiv:2305.15171*, 2023.

[52] R. Wu, B. Mildenhall, P. Henzler, K. Park, R. Gao, D. Watson, P. P. Srinivasan, D. Verbin, J. T. Barron, B. Poole *et al.*, "Reconfusion: 3d reconstruction with diffusion priors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 551–21 561.

[53] C. Yang, S. Li, J. Fang, R. Liang, L. Xie, X. Zhang, W. Shen, and Q. Tian, "Gaussianobject: Just taking four images to get a high-quality 3d object with gaussian splatting," *arXiv preprint arXiv:2402.10259*, 2024.

[54] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 22 500–22 510.

[55] P. Cao, F. Zhou, Q. Song, and L. Yang, "Controllable generation with text-to-image diffusion models: A survey," *arXiv preprint arXiv:2403.04279*, 2024.

[56] H. K. Cheng, S. W. Oh, B. Price, A. Schwing, and J.-Y. Lee, "Tracking anything with decoupled video segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 1316–1326.

[57] K. Liu, F. Zhan, J. Zhang, M. XU, Y. Yu, A. El Saddik, C. Theobalt, E. Xing, and S. Lu, "Weakly supervised 3d open-vocabulary segmentation," in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 53 433–53 456. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/file/a76b693f36916a5ed84d6e5b39a0dc03-Paper-Conference.pdf

[58] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," *arXiv preprint arXiv:2303.05499*, 2023.

[59] L. Ke, M. Ye, M. Danelljan, Y. Liu, Y.-W. Tai, C.-K. Tang, and F. Yu, "Segment anything in high quality," in *NeurIPS*, 2023.

[60] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," in *European Conference on Computer Vision*, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:1356654

[61] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv:2007.08501*, 2020.

[62] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–13, 2017.

[63] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik, "Lerf: Language embedded radiance fields," in *ICCV*, 2023.

[64] A. Haque, M. Tancik, A. Efros, A. Holynski, and A. Kanazawa, "Instruct-nerf2nerf: Editing 3d scenes with instructions," in *ICCV*, 2023.

[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:6628106

[66] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5855–5864.

Lizhi Wang received his bachelor's degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2023. His research interests include computer vision, and machine learning, especially focusing on 3d vision.

Feng Zhou received his bachelor's degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2022, and is currently a Ph.D. candidate at the School of Artificial Intelligence, Beijing University of Science and Technology. His research interests include computer vision, and machine learning, especially focusing on 3d vision.

Bo Yu received his bachelor's degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2023. His research interests include computer vision, and machine learning, especially focusing on multimodal generation.

Pu Cao received his bachelor's degree from University of Science and Technology Beijing, Beijing, China, in 2022, and is currently a Ph.D. candidate at the School of Artificial Intelligence, Beijing University of Science and Technology. His research interests include computer vision, and machine learning, especially focusing on multimodal generation.

Jianqin Yin received the Ph.D. degree from Shandong University, Jinan, China, in 2013. She currently is a Professor with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China. Her research interests include service robot, pattern recognition, machine learning and image processing. Email: jqyin@bupt.edu.cn.